1.

So good afternoon everybody, i hope i was able to wake you up a little? You’re probably thinking why would we start with such a song? Well, there are 2 reasons, as you can see here in the orange the text they sang was. “everything is awesome , everything is cool when you’re part of a team. This is something that reflects our project group very well and i will tell more about it in a moment.

2.

My name is Daniel Swaab, i swas ofcourse part of the Search and Rescue mission contrext project. I’ll show you where we started, what our begin situation was. Then i’ll go in on some refactoring that took was a big part of our project and after that you’ll see what we created (if you hve any qustions, please remember them until the end of the presentation?)

3.

The setup of our context project was a little different then yours. We are not the best team, not the worst, but we are one team. We worked with 19 students on this project. It was really cool and awesome to work in a team with this many students and to be able to learn and experience this. This is also the first reason (see the slide) for the song.

But why our we working with such a large team? We didn’t start from scratch. We got an excisting code base of the BW4T software. BW4T stands for ‘Block world for teams. It is software that interacts with a system/language called goal and visualizeses the actions of bots that have a mission in an environment and have to work togehter. An example of this is for instance war area where bot’s have to search for survivors or dead bodys. Many of you probably saw this software last year in the Multi agent Systems course. You see hre an example of how the visualization looks like. You see that everything is displayed as blocks, this is also the second reason for the song, the song was created for lego the movie, that is a movie about lego-blocks that have to work togheter.

4.

Our project had in important sub-title: ‘Crisis managment’ We started with 5500 lines of code, and no test and a very high coupling. As said before a part of the Project was to Refactor this system. As example how the code was structered i want to show you this uml diagram. There wasn’t a real clear structure. The features of the program are about what you see here. The bot has to deliver specified blocks to the dropzone, in an order, a block can pick up a block and walk. Now Suppose you are a bunch of researches in this specific domain. I mean – what would be the first thing you’d want to have made available for use? Wouldn’t it be nice to be able to customize your robots a little, instead of having this boring plain robot that runs around like an idiot? A few examples: Robot’s

* will jam/fail at some point.
* can’t walk on the same place at the same time
* Have different sizes
* Have different functionality
* The place can be really different, so you want an easy way to adjust the map etc.

These are things you want to test in the program as researcher but aren’t available. The other part of our project was to extends the software with these kind of new functionality.

So, now that’s clear, let start with the refactor part.

5.

We started with exploring the code, because it is really hard to extend a project if you have no test to check if your extenstions change anything and you have no idea where to how the code is structured. So we started looking into the code, and we came across interessting things, like an empty method, or an attribute that has a hack. We soon decided to create our own class diagramm, because the automatic created one wasn’t really clear. So we went throug classes and draw them on an board. Here we decided to restructure the complete system in 3 parts. The Server, Client and Core. By dividing all classes over these 3 parts, we could make sure that the server and client where independetly and this was also much better for testing. This is in what it resulted. You can see the different systems clear and that there are no direct connection between the server and client.

6.

To best way for us to show improvements in refactoring is by metrics. We installed sonar that shows a bunch of metrics about or system and by comparing the legacy to the new we can see our improvements. Also keep in mind that in the end we finished with 3 times the number of lines of where we started.

7.